CLAIMS

I Claim:

1. A storage system with data recovery from M failed blocks per stripe or J failed storage units comprising N (N>0) data blocks stored on N storage units and a first error correction code that generates M (M>0) redundant blocks from the N data blocks where the N data blocks and M redundant blocks form a stripe such that K (K less than or equal to M) blocks are regenerated from the remaining N+M-K blocks of the stripe, where the M redundant blocks are stored on J (J<M) additional storage units.
2. The storage system with data recovery from M failed blocks per stripe or J failed storage units of claim 1 wherein the storage for the redundant blocks is rotated among the N+J storage units such that the storage requirement is evenly distributed.
3. The storage system with data recovery from M failed blocks per stripe or J failed storage units of claim 1 wherein the storage for the redundant blocks is rotated among the N+J storage units such that the storage accesses are more evenly distributed.
4. The storage system with data recovery from M failed blocks per stripe or J failed storage units of claim 1 wherein the storage system provides additional data recovery from J failed storage units and L failed blocks per stripe where L (L less than or equal to M) redundant blocks that are copies of the M redundant blocks of a stripe are stored on the storage unit with the most recent data block update for the stripe and in the event of failure of storage units with the M redundant blocks, the L copies of the redundant blocks are used to reconstruct up to L failed blocks of the stripe.
5. The storage system with data recovery from M failed blocks per stripe or J failed storage units of claim 1 wherein the storage system provides additional data recovery from J failed storage units and L failed data blocks per stripe where L (L less than or equal to M) redundant blocks that are copies of the M redundant blocks of a stripe are stored on the storage unit with the most recent data block update for

the stripe and the storage blocks for the L copies of the redundant blocks are assigned as needed from a pool of storage blocks.

6. The storage system with data recovery from M failed blocks per stripe or J failed storage units of claim 1 wherein the storage system provides additional data recovery from J failed storage units and L failed blocks in the stripe and R failed blocks for each stripe of a second stripe structure within each functioning storage unit where
   - L (L less than or equal to M) redundant blocks that are copies of the M redundant blocks of a stripe are stored on the storage unit with the most recent data block update for the stripe,
   - For S blocks stored on a storage unit including one block from the stripe, a second error correction code generates R redundant blocks from the S blocks such that V (V less than or equal to R) blocks are regenerated from the remaining S+R-V blocks of the second stripe and the R redundant blocks are stored on the storage unit.
7. The storage system with data recovery from M failed blocks per stripe or J failed storage units of claim 1 wherein the storage system provides additional data recovery from J failed storage units and R failed blocks for each stripe of a second stripe structure in each functioning storage unit where
   - For S blocks stored on a storage unit including one block from the stripe, a second error correction code generates R redundant blocks from the S blocks such that V (V less than or equal to R) blocks are regenerated from the remaining S+R-V blocks of the second stripe and the R redundant blocks are stored on the storage unit,
8. A storage system with data recovery from L failed blocks per stripe comprising N (N>0) data blocks stored on H (H>0) storage units and a first error correction code that generates M (M>0) redundant blocks from the N data blocks where the N data blocks and the M redundant blocks form a stripe such that K (K less than or equal to M) blocks are regenerated from the remaining N+M-K blocks of the stripe and L (L less than or equal to M) redundant blocks are stored on the storage unit with the most recent data block update such that T (T less than or equal to L) blocks are regenerated from the remaining N+L-T blocks of the stripe.

9. The storage system with data recovery from L failed blocks per stripe of claim 8 wherein the storage blocks for the L redundant blocks are assigned as needed from a pool of storage blocks.
10. The storage system with data recovery from L failed blocks per stripe of claim 8 wherein the storage system provides additional data recovery from J failed storage units or M failed blocks per stripe where the number of data blocks, N, equals the number of storage units, H, each with a data block from the stripe and the M redundant blocks for the stripe are stored on J (J less than or equal to M) additional storage units.
11. The storage system with data recovery from L failed blocks per stripe of claim 8 wherein the storage system provides additional data recovery from J failed storage units or M failed blocks per stripe where the number of data blocks, N, equals the number of storage units, H, each storing a data block from the stripe, and the M redundant blocks for the stripe are stored on J (J less than or equal to M) additional storage units and the storage requirement for the M redundant blocks is rotated among the H+J storage units so the storage requirement is equally distributed.
12. The storage system with data recovery from L failed blocks per stripe of claim 8 wherein the storage system provides additional data recovery from J failed storage units or M failed blocks per stripe where the number of data blocks, N, equals the number of storage units, H, each storing a data block from the stripe, and the M redundant blocks for the stripe are stored on J (J less than or equal to M) additional storage units and the storage requirement for the M redundant blocks is rotated among the H+J storage units so the storage accesses are more evenly distributed
13. The storage system with data recovery from L failed blocks per stripe of claim 8 wherein the storage system provides additional data recovery from J failed storage units and L failed blocks per stripe and R failed blocks per second stripe within a storage unit or M failed blocks per stripe and R failed blocks per second stripe within a storage unit where
    - The number of data blocks, N, equals the number of storage units, H, each storing a data block from the stripe,
    - The M redundant blocks for the stripe are stored on J (J less than or equal to M) additional storage units and

- For S blocks stored on a storage unit including one block from the stripe, a second error correction code generates R redundant blocks from the S blocks such that V (V less than or equal to R) blocks are regenerated from the remaining S+R-V blocks of the second stripe and the R redundant blocks are stored on that storage unit.

14. A storage system with data recovery from R failed blocks per second stripe within a storage unit and J failed storage units or M failed blocks per first stripe across storage units and R failed blocks per second stripe within a storage unit comprising
    - N (N>0) data blocks stored on N storage units and
    - A first error correction code that generates M (M>0) redundant blocks from the N data blocks where the N data blocks and M redundant blocks form a first stripe across storage units such that K (K less than or equal to M) blocks are regenerated from the remaining N+M-K blocks of the first stripe and
    - The M redundant blocks are stored on J (J less than or equal to M) additional storage units and
    - S blocks stored on a storage unit including one block from the first stripe and a second error correction code that generates R (R>0) blocks from the S data blocks where the S blocks and R redundant blocks form a second stripe within the storage unit such that V (V less than or equal to R) blocks are regenerated from the remaining S+R-V blocks of the second stripe and
    - The R redundant blocks are stored on that storage unit.
15. The storage system with data recovery from R failed blocks per second stripe within a storage unit and J failed storage units or M failed blocks per first stripe across storage units and R failed blocks per second stripe within a storage unit of claim 14 wherein the storage requirement for the M redundant blocks is rotated among the N+J storage units so that the storage requirement is evenly distributed.
16. The storage system with data recovery from R failed blocks per second stripe within a storage unit and J failed storage units or M failed blocks per first stripe across storage units and R failed blocks per second stripe within a storage unit of claim 14 wherein the storage requirement for the M redundant blocks is rotated among the N+J storage units so that the storage accesses are more evenly distributed.

17. The storage system with data recovery from R failed blocks per second stripe within a storage unit and J failed storage units or M failed blocks per first stripe across storage units and R failed blocks per second stripe within a storage unit of claim 14 wherein the storage system provides additional data recovery from R failed blocks per second stripe and L failed blocks per first stripe across the storage units and J failed storage units where L (L less than or equal to M) redundant blocks are L copies of the M redundant blocks are stored on the storage unit with the most recent data block update.
18. The storage system with data recovery from R failed blocks per second stripe within a storage unit and J failed storage units or M failed blocks per first stripe across storage units and R failed blocks per second stripe within a storage unit of claim 14 wherein the storage system provides additional data recovery from R failed blocks per second stripe and L failed blocks per first stripe across the storage units and J failed storage units where L (L less than or equal to M) redundant blocks are L copies of the M redundant blocks are stored on the storage unit with the most recent data block update wherein the storage blocks for the L redundant blocks are assigned on demand from a pool of storage blocks.